

IFT 436 – Algorithmes et structures de données
Devoir #1 : Combinaison de tri par insertion et tri par fusion
Date de remise : le lundi 21 septembre 2015

Les buts de ce devoir sont multiples. D'une part, il vise à combiner deux algorithmes de tri pour créer un nouvel algorithme permettant de profiter des avantages des deux algorithmes. D'autre part, il vise aussi à se familiariser avec l'analyse de complexité aussi bien sur le plan théorique que pratique. Finalement, ce premier devoir vise aussi une calibration au niveau de l'exigence quant à la manière et la qualité de présentation de devoirs.

Description du devoir : Bien que, dans le pire des cas, le tri par fusion s'exécute en $\theta(n \lg(n))$ et le tri par insertion en $\theta(n^2)$. Les facteurs constants du tri par insertion peuvent le rendre plus rapide pour de petites tailles de problème sur des machines. Il est donc logique de rendre plus grossières les feuilles de la récursivité en utilisant le tri par insertion à l'intérieur du tri par fusion, lors que les sous-problèmes deviennent suffisamment petits. Supposons que cette « petite » taille soit k , nous aurons n/k sous-listes de longueur k à trier par insertion. Les résultats de ces listes triées doivent être fusionnés par le mécanisme de tri-par-fusion. La valeur k est à déterminer.

- 1- Présenter l'algorithme modifié (algorithme combiné) en pseudo code.
- 2- Montrer que les n/k sous-listes, chacune de longueur k , peuvent être triées via tri par insertion avec un temps de $\theta(nk)$ dans le cas le plus défavorable.
- 3- Montrer que les sous-listes peuvent être fusionnées en $\theta(n \lg(n/k))$ dans le cas le plus défavorable.
- 4- Sachant que l'algorithme modifié s'exécute en $\theta(nk + n \lg(n/k))$ dans le cas le plus défavorable, quelle est la plus grande valeur de k , en tant que fonction de n , pour laquelle l'algorithme modifié a le même temps d'exécution asymptotique que le tri par fusion classique? Pour répondre à cette question, vous pouvez supposer que lors que $\theta(g(n))$, vous avez $\theta(g(n)) = ag(n)$ avec la constante $a > 0$ connue mais spécifique à cette $g(n)$.
- 5- Déterminer la valeur de k expérimentalement. Cette valeur dépend de la machine, donc vous devez faire des tests sur le même ordinateur. Une façon de faire est la suivante : 1) générer des listes de données les plus défavorables de différentes longueurs, e.g. $n = 10^2, 10^3, 10^4, 10^5, \dots$, ou $n = 10^6, 10^6 + 10^5, 10^6 + 2 * 10^5, 10^6 + 3 * 10^5, \dots$; 2) tester l'algorithme modifié pour de différentes valeurs de k et l'algorithme de tri par fusion classique afin de trouver la valeur de k telle que le temps de tri par l'algorithme modifié soit le même que celui de l'algorithme de tri par fusion; 3) Tracer une courbe de k en fonction de n .

Exigence pour la remise : Pour la remise du devoir, vous devez remettre un rapport, en format PDF, contenant des éléments suivants :

- Une page de couverture contenant le(s) nom(s) de l'équipe et le(s) numéro(s) de matricule. Rappelons que les devoirs du cours IFT436 doivent être complétés par équipe d'une personne ou de deux personnes. Pour encourager l'autonomie dans la réalisation des travaux, 50% de points vous seront automatiquement accordés sur la remise d'un rapport contenant au moins cette page de couverture.
- Dans la présentation des solutions à chacun des questions ci-dessus, vous devez faire des efforts pour être rigoureux du point de vue mathématique. Le nombre de pages n'est pas limité, mais il est toujours préférable que la partie principale du rapport soit la plus brève possible.

- Pour la sous-question 5, vous présentez, dans l'ordre, votre méthode, la machine de test si vous l'avez fait sur un ordinateur du DI ou les caractéristiques principales de l'ordinateur de test tel que le modèle de processeur et la quantité de mémoire interne, et les résultats (surtout la courbe). À la fin, vous devez fournir aussi le code de vos programmes.

La remise doit être faite par ... à déterminer ...